

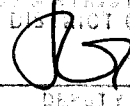
IN THE UNITED STATES DISTRICT COURT
FOR THE WESTERN DISTRICT OF TEXAS
AUSTIN DIVISION

FILED

2015 MAR 19 AM 9:27

CLERK OF DISTRICT COURT
WESTERN DISTRICT OF TEXAS

BY


DEPUTY

FISHER-ROSEMOUNT SYSTEMS, INC.,
Plaintiff,

-vs-

Case No. A-13-CA-587-SS

INVENSYS SYSTEMS, INC.,
Defendant,

and

INVENSYS SYSTEMS, INC.,
Counter-Plaintiff,

-vs-

FISHER-ROSEMOUNT SYSTEMS, INC., and
EMERSON ELECTRIC CO.,

Counterclaim-Defendants.

ORDER

BE IT REMEMBERED on this day the Court reviewed the file in the above-styled cause, specifically Plaintiff Fisher-Rosemount Systems, Inc. and Emerson Electric Co. (collectively, Fisher-Rosemount)'s Opening Claim Construction Brief [#65], Defendant Invensys Systems, Inc. (Invensys)'s Opening Claim Construction Brief [#64], Fisher Rosemount's Corrected Reply Claim Construction Brief [#70-1], Invensys's Reply Claim Construction Brief [#68], Fisher-Rosemount's Opening Post-*Markman* Brief [#74], Invensys's Opening Post-*Markman* Brief [#75], Fisher-Rosemount's Responsive Post-*Markman* Brief [#76], Invensys's Responsive Post-*Markman* Brief [#77], the parties' Joint Chart Identifying Disputed Claim Terms [#78], Fisher-Rosemount's Sur-

✓

Reply [#83], the Report and Recommendation of the Special Master [#84], Fisher-Rosemount's Objections [#87], Invensys's Objections [#88], Fisher-Rosemount's Response to Invensys's Objections [#91], and Invensys's Response to Fisher-Rosemount's Objections [#92]. Having reviewed the documents, the governing law, the arguments of the parties at the *Markman* hearing, and the file as a whole, the Court now enters the following opinion and orders.

Background

I. Procedural History

Plaintiff Fisher-Rosemount filed this patent infringement lawsuit against Defendant Invensys, asserting nine patents. At this time, the parties disagree regarding the construction of three claim terms that appear in two of those nine patents: U.S. Patent Nos. 7,043,311 (the '311 Patent) and 7,971,052 (the '1052 Patent). Invensys filed counterclaims for infringement of seven of its patents. Currently, the parties disagree regarding the construction of five claim terms that appear in five of those seven patents: U.S. Patent Nos. 6,799,195 (the '195 Patent); 7,502,656 (the '656 Patent); 7,720,944 (the '944 Patent); 7,899,070 (the '070 Patent); and 7,739,361 (the '361 Patent).

The Court, through Special Master Karl Bayer, held the *Markman* hearing on June 24, 2014. The Special Master issued his Report and Recommendation on claim construction on September 29, 2014. To the extent the parties have made specific objections to the Special Master's factual findings or legal conclusions, they are entitled to de novo review of those findings and conclusions. FED. R. CIV. P. 53(f).

II. Patent Descriptions

All of the asserted patents generally relate to various aspects of process automation management and process control systems. Process automation management concerns developing

ways for engineers, floor operators, and others working in a manufacturing facility to monitor and maintain plant operating conditions so the manufactured product meets the desired specifications while increasing safety, reducing costs, and decreasing energy consumption, among other benefits. Within a process automation management system, specific process control functions are carried out using software programs that run in the computers and controllers. In configuring any facility, plant engineers write software programs to control and direct the many functions that must be carried out in the plant.

Fisher-Rosemount's patents claim better ways to configure and operate process automation management systems. The '311 Patent discloses the use of object-oriented programming concepts in process automation management systems. The '311 Patent creates efficiencies for plant engineers by teaching the use of "module class objects" in process automation management systems with the idea behind module class objects being that engineers can create generic control module classes for control activity throughout the plant.

The '1052 Patent discloses and claims the use of security control and builds on the object-oriented programming principles disclosed in the '311 Patent. In particular, the '1052 Patent discloses the ability to restrict or allow changes to a particular object and the further ability to determine whether changes will apply to related objects throughout the plant. Prior art systems had no way of controlling what certain operators or users could see or access, but the '1052 Patent teaches that access to objects through the plant can be controlled by associating user access control data with the object. The user access control data indicates whether a given user can view or modify data associated with a given object within the plant environment.

Invensys's patents generally relate to the application of computing technology for the purpose of remotely accessing process control systems. Prior art process control systems were primarily designed to be accessed by users within the plant using the plant's network. Invensys's patents used computing technology, including (1) information clients, (2) platform-independent source code languages, (3) virtual machines, (4) information clients that include Java virtual machines, (5) applets, and (6) object managers to allow for remote access. The '656 Patent is directed to accessing a process control system over a wireless network using portable computing devices. Invensys's other four patents—the '195 Patent, '944 Patent, '361 Patent, and the '070 Patent—are directed to accessing a process control system using applets in a virtual machine environment.

Analysis

I. Claim Construction—Legal Standard

When construing claims, courts begin with “an examination of the intrinsic evidence, i.e., the claims, the rest of the specification and, if in evidence, the prosecution history.” *CCS Fitness, Inc. v. Brunswick Corp.*, 288 F.3d 1359, 1366 (Fed. Cir. 2002); *see also Interactive Gift Express, Inc. v. Compuserve Inc.*, 256 F.3d 1323, 1327 (Fed. Cir. 2001).

The words in the claims themselves are of primary importance in the analysis, as the claim language in a patent defines the scope of the invention. *SRI Int'l v. Matsushita Elec. Corp.*, 775 F.2d 1107, 1121 (Fed. Cir. 1985) (en banc). The words of a claim “are generally given their ordinary and customary meaning.” *Phillips v. AWH Corp.*, 415 F.3d 1303, 1312 (Fed. Cir. 2005). “[T]he ordinary and customary meaning of a claim term is the meaning that the term would have to a person of ordinary skill in the art in question at the time of the invention, i.e., as of the effective filing date of

the patent application.”¹ *Id.* at 1313. The inquiry into how a person of ordinary skill in the art understands a claim term provides an “objective baseline” from which to begin claim interpretation. *Id.* The person of ordinary skill in the art is understood to read a claim term not only in the context of the particular claim in which the term appears, but in the context of the entire patent, including the specification; thus, both the plain language of the claims and the context in which the various terms appear “provide substantial guidance as to the meaning of particular claim terms.” *Id.* at 1314.

The specification also plays a significant role in the analysis. *Id.* at 1315. The Federal Circuit has repeatedly reaffirmed the principle that the specification “is always highly relevant Usually, it is dispositive; it is the single best guide to the meaning of a disputed term.” *Id.* (quoting *Vitronics Corp. v. Conceptronic, Inc.*, 90 F.3d 1576, 1582 (Fed. Cir. 1996)). In interpreting the effect the specification has on the claim limitations, however, courts must pay special attention to the admonition that one looks “to the specification to ascertain the meaning of the claim term as it is used by the inventor in the context of the entirety of his invention, and not merely to limit a claim term.” *Interactive Gift*, 256 F.3d at 1332 (internal quotation marks and citations omitted).

The final form of intrinsic evidence the Court may consider is the prosecution history. Although the prosecution history “represents an ongoing negotiation between the PTO and the applicant” and therefore “often lacks the clarity of the specification and thus is less useful for claim construction purposes,” it can nonetheless “often inform the meaning of the claim language by demonstrating how the inventor understood the invention and whether the inventor limited the

¹ This hypothetical person is now commonly referred to simply as an “ordinarily skilled artisan.” *E.g.*, *Power Integrations, Inc. v. Fairchild Semiconductor Int’l, Inc.*, 711 F.3d 1348, 1365–66 (Fed. Cir. 2013).

invention in the course of prosecution, making the claim scope narrower than it would otherwise be.”

Phillips, 415 F.3d at 1317.

Aside from the intrinsic evidence, the Court may also consult “extrinsic evidence,” which is “all evidence external to the patent and prosecution history, including expert and inventor testimony, dictionaries, and learned treatises.” *Id.* While extrinsic evidence “can shed useful light on the relevant art,” the Federal Circuit has explained it is “less significant than the intrinsic record in determining ‘the legally operative meaning of claim language.’” *Id.* at 1317 (quoting *C.R. Bard, Inc. v. U.S. Surgical Corp.*, 388 F.3d 858, 862 (Fed. Cir. 2004)). Extrinsic evidence in the form of expert testimony may be useful to a court for “a variety of purposes, such as to provide background on the technology at issue, to explain how an invention works, to ensure that the court’s understanding of the technical aspects of the patent is consistent with that of a person of skill in the art, or to establish that a particular term in the patent or the prior art has a particular meaning in the pertinent field.” *Id.* at 1318. However, conclusory, unsupported assertions by an expert as to the definition of a claim term are not useful, and should be discounted. *Id.* In general, extrinsic evidence is considered “less reliable than the patent and its prosecution history in determining how to read claim terms,” although it may be helpful. *Id.*

The purpose of claim construction is to “‘determin[e] the meaning and scope of the patent claims asserted to be infringed.’” *02 Micro Int’l Ltd. v. Beyond Innovation Tech. Co.*, 521 F.3d 1351, 1360 (Fed. Cir. 2008) (quoting *Markman v. Westview Instruments, Inc.*, 52 F.3d 967, 976 (Fed. Cir. 1995), *aff’d*, 517 U.S. 370 (1996)). Thus, “[w]hen the parties raise an actual dispute regarding the proper scope of these claims, the court, not the jury, must resolve that dispute.” *Id.* However, “district courts are not (and should not be) required to construe *every* limitation present in a patent’s

asserted claims.” *Id.* at 1362. For example, no construction is required if the requested construction would be “an obligatory exercise in redundancy,” or if the disputed issue is the proper application of a claim term to an accused process rather than the scope of the term. *Id.* (citations omitted).

II. Application

A. Special Master’s Recommendations

The Special Master’s recommended constructions are as follows:

Claim Term	Recommended Construction
“track the module class object from which each of the one or more module objects was created” (’311 Patent, claims 27, 44, and 45)	Indefinite.
“storing an indication of the module class object from which each of the one or more module objects was created” (’311 Patent, claims 54, 70, 71, and 77)	No construction necessary.
“set[ting] a value of a parameter included in the object to indicate that access to the object by a viewing or configuration application is to be limited” (’1052 Patent, claims 5, 9, and 21)	No construction necessary.
“information client” (’195 Patent, claims 1, 14, and 16; ’944 Patent, claims 4 and 7; ’361 Patent, claim 7)	No construction necessary.
“information server” (’361 Patent, claim 7)	No construction necessary.
“virtual machine environment” (’195 Patent, claims 1, 14, and 16; ’944 Patent, claims 4 and 7; ’361 Patent, claims 3, 7, 12, and 13; ’070 Patent, claims 3 and 7)	No construction necessary.

“digital data processor is disposed remotely from the portable computer” (’656 Patent, claim 13)	No construction necessary.
“JAVA/API application for transferring information between the JAVA application and the API” (’944 Patent, claim 15)	No construction necessary.

B. Objections

The Court now turns to the parties’ specific objections.

1. “track the module class object from which each of the one or more module objects was created”

The Special Master recommended this term be construed as indefinite. Definiteness is to be evaluated from the perspective of someone skilled in the relevant art. *Nautilus v. Biosig Instruments Inc.*, 134 S. Ct. 2120, 2128 (2014). In assessing definiteness, claims are to be read in light of the patent’s specification and prosecution history. *Id.* Definiteness is measured from the viewpoint of a person skilled in the art at the time the patent was filed. *Id.* The Supreme Court recently lowered the standard for indefiniteness in *Nautilus*. The previous standard assessed definiteness based on whether the claims were “amenable to construction” or “insolubly ambiguous.” *Id.* at 2130. The Court rejected these approaches because “it cannot be sufficient that a court can ascribe *some* meaning to a patent’s claims; the definiteness inquiry trains on the understanding of a skilled artisan at the time of the patent application, not that of a court viewing matters *post hoc*.” *Id.* (emphasis in original). Application of these standards “diminish[ed] the definiteness requirement’s public-notice function and foster[ed] the innovation-discouraging ‘zone of uncertainty,’ against which this Court has warned.” *Id.* (internal citation omitted). In place of the old standard, the Court adopted a new one: whether “[the] claims, read in light of the specification delineating the patent, and the

prosecution history, fail to inform, with reasonable certainty, those skilled in the art about the scope of the invention.” *Id.* at 2124.

The parties’ dispute centers on the meaning of the word “track” and, more precisely, the meaning of “track” in the context of the entire phrase at issue. The parties agree the term is not a term of art, and Invensys agrees the word “track” alone would not confuse the jury. *See Markman* Hr’g Tr. [#73] at 17:24–18:2, 22:13–14. The core problem is determining what it means to “track the module class object.” The phrase must inform with reasonable certainty those skilled in the art as to the scope of the invention. The term, however, only appears once in the patent—in claim 27—and is totally absent from the specification. *Id.* at 32:11–13 (Special Master: “Do you agree, though, that track is not used anyplace but the claim?” Fisher-Rosemount’s counsel: “Yeah. That is true.”). As a result, there is essentially no explanation for the meaning of the word “track” or the meaning of the phrase “track the module class object from which each of the one or more class objects was created.”

The specification generally describes an invention whereby a module class object (e.g., reactors) is used to designate a generic class of equipment, and within the class of reactors, specific reactors are identified as module objects (e.g., reactor 1, reactor 2, etc.). This hierarchical structure allows programmers to make changes at one level (the module class object level) that are propagated to specific instances on the plant floor (the module objects). Given this structure, it is not clear what it means to “track the module class object.” If anything, it makes sense to track the module objects. *See* ’311 Patent, col. 19 ll. 41–47 (disclosing the maintenance of a list of the module objects). Yet claim 27 specifically provides “track the module class object,” and the patent simply does not explain what this phrase means.

Another sensible concept with a foundation in the patent would be tracking the relationship between the module class object and the module object. Indeed, Fisher-Rosemount tries to give the phrase this meaning by emphasizing how the specification describes the creation and programming of module class objects and module objects, and the importance of their relationship to one another. Pl.'s Objections [#87] at 4. Therefore, according to Fisher-Rosemount, "the only possible meaning of this phrase is that the relationship between the module class object and the associated module objects is being tracked. For the seamless propagation described in the patent to work, the claimed invention needs to know (or 'track') the relationships between module class objects and module objects." *Id.* In other words, Fisher-Rosemount tries to equate "track the module class object from which each of the one or more module objects was created" with tracking the *relationship* between module class objects and module objects. But these phrases are not equivalent. Tracking the relationship between "reactors" as the module class object and "reactor 1" and "reactor 2" as the module objects is a distinct concept from tracking the module class object "reactors." If the phrase at issue did indeed use "track the relationship" language, the phrase would be more definite, but the key point is the phrase does not use that language. Instead, the language is "track the module class object," and Fisher-Rosemount's attempt to transform this language into meaning to "track the relationship" between module class objects and module objects only illustrates the indefiniteness of the claim as written.

Beyond the confusion surrounding what it means to "track the module class object," it is also unclear what the single word "track" means given the lack of explanation anywhere in the patent. As Invensys suggests, "to track" has multiple possible meanings, including to "monitor the changes of," to "store the edits to," to "mimic," to "gather the parameters of," to "list the composition of,"

to “identify the version of,” or to “identify the relationship(s) of.” *Markman* Hr’g Tr. [#73] at 18:13–20:14; Def.’s Opening Post-*Markman* Hr’g Br. [#75-4] (Second Stephenson Decl.) ¶ 29. Nothing in the specification clarifies the ambiguity of “track,” which adds a layer of uncertainty to an already unclear phrase “track the module class object.”

The following exchange at the *Markman* hearing demonstrates the patent’s failure to inform, with reasonable certainty, those skilled in the art of the scope of the “track” phrase:

Special Master: Okay. So if I’m somebody, then, out there in the real world and I’m trying not to step on your patent . . . and I don’t want to keep track of it in the way you’re keeping track of it, how do I know that? How do I know that from reading anywhere in the patent or looking at a figure, regardless of whether we call it indefiniteness, enablement, any of those things? How do I avoid stepping on your patent, sort of substituting the common phrase “keeping track of” for the word “track”?

Fisher-Rosemount’s counsel: Well, I think you know from the full context of the specification and the rest of the—I think you’ve got to look at everything.

Special Master: Well, okay. So tell me, then, look at everything for me because that’s a lot.

Fisher-Rosemount’s counsel: The context of the specification, the claim and—

Special Master: Well, what—okay. So again, that all seems sort of nebulous to me. Show me where in the specification or the claim I know you’re keeping track of things.

Fisher-Rosemount’s counsel: I’m not—I think I have to consider that to give you a complete answer.

Special Master: Uh-huh.

Fisher-Rosemount’s counsel: And I’m not sure I can do that—

Special Master: Okay.

Fisher-Rosemount’s counsel: —as I stand here.

Special Master: All right. Fair enough.

Markman Hr'g Tr. [#73] at 14:8–15:9.

In sum, the phrase “track the module class object from which each of the one or more module objects was created,” when read in the context of the patent, fails to inform, with reasonable certainty, those skilled in the art about the scope of the invention. To give it reasonably certain meaning, Fisher-Rosemount asks the Court to rewrite the phrase to read “track the relationship between the module class object and the module objects,” but rewriting claim language is beyond this Court’s task. *See Allen Eng’g Corp. v. Bartell Indus., Inc.*, 299 F.3d 1336, 1349 (Fed. Cir. 2002) (“It is not our function to rewrite claims to preserve their validity.”). Moreover, the meaning of “track,” when read in the context of the phrase, “track the module class object from which each of the one or more module objects was created,” is unclear, as it is open to many plausible meanings. And it is not sufficient that a court can ascribe some meaning to the word “track.” Instead, the word must inform, with reasonable certainty, a skilled artisan about the scope of its meaning, and here, the patent fails.

The Special Master was correct to recommend the claim term “track the module class object from which each of the one or more module objects was created” be construed as indefinite. Fisher-Rosemount’s objection is OVERRULED.

2. “storing an indication of the module class object from which each of the one or more module objects was created”

The Special Master recommended that construction is unnecessary for this claim phrase. Invensys objects to this recommendation, contending the phrase should either be construed as indefinite or, alternatively, as “storing a list or other indication of each of the one or more module

objects created from the module class object.” In other words, Invensys argues the specification contemplates an invention directed to using module class objects (which Invensys likes to refer to as “parents”) to create and own module objects (which Invensys likes to refer to as “children”), and therefore the patent can only be read to encompass storing indications of the module objects, not the module class objects.

The specification, however, does include language suggesting references to the module class objects—not just module objects—are stored. Specifically, the specification includes the following paragraph, which is hotly debated between the parties:

Moreover, the unit module class object **410** may include a memory or section **526** that stores references to the module class objects which are created by the configuration application **50** (FIG. **1**) from this unit module class object **410**. The section **526** is essentially a list of the module objects that were created from and that are owned by the unit module class object **410**. (Of course this list or other indication of the owned module objects can be stored in the workstation or by the configuration application **50** in any desired manner and does not have to be physically contained in the unit module class object **410**). In any event, in the example of FIG. **6**, the unit module class object **410** owns module class objects Reactor_01, Reactor_1, Reactor_02, etc., each of which has been created from the reactor unit module class object **410**.

’311 Patent, col. 19 ll. 40–56.

Here, the specification clearly states “the unit module class object **410** may include a memory or section **526** that *stores references to the module class objects . . .*” *Id.* (emphasis added). Invensys’ primary defense to this language is that it must be a typographical error, and “module objects” should replace “module class objects” in this sentence. *See Markman* Hr’g Tr. [#73] at 34:4–9. According to Invensys, this language must be a mistake because it suggests parents storing references to other parents when the patent only covers parents storing references to their children. *Id.*

Invensys's position is misplaced for a few reasons. First, while the only supposed typographical error Invensys directly highlights is in the above-quoted paragraph in the specification, it implicitly requires the Court to conclude the claim language itself, which clearly states "storing an indication of the module class object from which each of the one or more module objects was created," is also a typographical error. In other words, Invensys's proposed construction asks the Court to rewrite the claim language from "storing an indication of the module class object" to "storing a list or other indication of each of the one or more module objects," which are obviously two different concepts. Rewriting claims into completely different terms is not within the Court's claim construction function.

Second, Invensys's conclusion the above-quoted paragraph is a typographical error (or at least ambiguous) stems from its reliance on a rigid parent-child paradigm. Yet the patent contemplates a more nuanced system of relationships than simply parents creating children. As an initial matter, Invensys fails to account for the word "unit" in reading the above-quoted paragraph. So instead of "parents" storing references to "parents," the language actually describes the "unit parent" storing references to the "parent" which are created by the configuration application from this "unit parent."

In addition, the patent describes many different possible relative relationships (and interrelationships) between module class objects and module objects. For example, the patent covers module class objects incorporating other module class objects:

Each module class object may have sub-objects associated therewith or owned thereby. These sub-objects may be module class objects in themselves or, as illustrated in FIG. 5, may be module objects which are created as instances of the module class object to which they belong.

'311 Patent, col. 16 ll. 14–18.

The patent also indicates module class objects defining other module class objects:

As part of a module class object definition, a module class object may indicate or define other module class objects to be incorporated or used therein. When this is the case, the module objects created from that module class object will incorporate, refer to or include other module objects created from other module class objects according to the relationships defined at the module class level. Although not strictly necessary, unit module class objects may incorporate other unit module class objects, equipment module class objects, control module class objects and display module class objects. . . .

Id., col. 17 ll. 26–38.

Accordingly, the specification contemplates a programming scheme that is not only vertically integrated but also horizontally integrated between module class objects and the associated module objects. Given the multiple levels and interrelationships described in the patent, Invensys's simple parent-child paradigm is inadequate, and reliance on this inflexible structure leads to ambiguity, confusion, and suggestions of typographical error when applied to the patent's language.

Third, Invensys argues that because the patent repeatedly describes the invention in the context of module class objects creating module objects, then there is no support for the idea of storing indications of module class objects. *See* Invensys's Objections [#88] at 9–11 (citing a number of examples from the patent describing the creation of module objects from module class objects). The fact module objects are created from module class objects, however, is not in dispute. Nor does this fact suggest the patent does not describe storing indications of module class objects or that "storing" is limited to module objects. To the contrary, the creation of module objects from module class objects and the storing of indications of the module class objects from which each of the module objects was created are compatible ideas, both of which are described in the patent.

In sum, to the extent Invensys argues the claim phrase “storing an indication of the module class object from which each of the one or more module objects was created” is indefinite, the Court disagrees. Invensys concedes “storing an indication” has a well-understood meaning to a person of ordinary skill in the art. *See* Def.’s Opening Claim Construction Br. [#64] at 23. And the plain language of the claim is clear when it states “storing an indication of the module class object,” and this language is repeated and explained in the specification. The claim informs, with reasonable certainty, those skilled in the art about the scope of the invention, and therefore it is not indefinite under *Nautilus*. Moreover, to the extent Invensys urges its alternative construction, the Court rejects this attempt to completely rewrite the claim language and the specification. Contrary to Invensys’s position, the patent clearly discusses storing indications of module class objects, and Invensys cites nothing in the patent limiting “storing” to module objects.

The Special Master was correct to recommend no construction is necessary, and Invensys’s objection is OVERRULED.

3. “set[ting] a value of a parameter included in the object to indicate that access to the object by a viewing or configuration application is to be limited”

The Special Master recommended that construction is unnecessary for this claim phrase. Invensys objects and argues the construction should include the word “directly” before “indicate.” The parties’ disagreement centers around interpretation of the ’1052 Patent’s prosecution history.

Originally, this limitation more broadly recited “associating with the object an indication that access to the object by a viewing or configuration is to be limited.” *See* Def.’s Opening Claim Construction Br. [#64-5] Ex. 5. The Patent Office rejected the claim based on the Eldridge prior art. *See id.* [#64-4] Ex. 4. To overcome the Eldridge prior art, the applicants narrowed the limitation

from “associating with an object an indication . . .” to “setting a value of a parameter included in the object to indicate . . .” *Id.* [#64-5] Ex. 5. The amendment specified the location of the parameter indicating access to the object; instead of simply being “associated” with the object, the parameter was “included in the object.” The Patent Office approved the claim as amended.

Despite this clear history, Invensys now contends the claim as amended does not avoid the Eldridge prior art. *See* Def.’s Opening Post-*Markman* Br. [#75] at 11. Instead, in order to avoid Eldridge, Invensys argues the claim must read “setting a value of a parameter included in the object to *directly* indicate . . .” The prosecution history does not support Invensys’s position. If the applicants sought to amend the claim by including the word “directly,” then they would have done so. Their focus, however, was on the location of the parameter, and Invensys cites no evidence in the prosecution history indicating any debate or discussion regarding whether the indication was direct or indirect. The Patent Office approved the amendment, presumably because it agreed the new language clarifying the location of the parameter avoided the Eldridge prior art.

Invensys disagrees and insists the Eldridge reference included the security characteristics “within the object.” *See id.* The prosecution history indicates otherwise. The applicants noted the Eldridge prior art had “security objects that define security access to objects based on users, groups, areas within a plant, and/or object types.” *See* Def.’s Opening Claim Construction Br. [#64-5] Ex. 5 at 8. The applicants also pointed out “[t]hese security objects, however, are distinct from objects that correspond to actual process entities used in process control routines . . .” *Id.* As an example, the applicants cited Eldridge’s FIG. 58 as showing “Eldridge’s security objects . . . are clearly separate and distinct from the box labeled ‘Assignable Object.’” *Id.* at 8–9. Distinguishing the amended claims, the applicants explained that “[i]n the present application, however, indication of

access to an object that corresponds to a process entity used in a process control routine is simply indicated by a value of a parameter *included in the object itself*.” *Id.* at 9 (emphasis added). Clearly, the basis for the applicant’s amendment was to distinguish the claims from Eldridge based on the location of the parameter. There was no discussion of whether the indication was direct or indirect.

The plain language of the claim phrase is straightforward, and the Special Master was correct to recommend no construction is necessary. Invensys’s objection is **OVERRULED**.

4. “information client” and “information server”

The Special Master recommended that construction is unnecessary for these terms. The parties agree the terms “client” and “server” were well-known in the art. Fisher-Rosemount argues these terms from the ’195 Patent are different in that they include the modifier “information,” and, according to Fisher-Rosemount, the patent provides specific definitions for them. Because Invensys supposedly acted as its own lexicographer, Fisher-Rosemount contends it should be bound by its own definitions.

Specifically, Fisher-Rosemount cites the following passage as reflective of lexicographical intent concerning “information client”:

Although a preferred information client is a web browser, the invention can be practiced with other information clients capable of (1) initiating communications with the information server **20**, (2) requesting and receiving from the information server **20** an applet, and (3) defining a platform-independent (i.e., a hardware-independent, operating system-independent and windows system independent) virtual machine environment within the respective client digital data processor **12, 14** for execution of such an applet.

’195 Patent, col. 5 ll. 37–45. Tracking this language, Fisher-Rosemount proposes the following construction for information client: “an application, such as a web browser, that can (1) initiate communications with an information server over a network; (2) request and receive an applet from

an information server; and (3) define a hardware-independent, operating system-independent and windows system independent virtual machine environment.”

Similarly, Fisher-Rosemount argues the patent defines “information server”:

The information server **20** is preferably a hypertext transfer protocol (HTTP) server capable of transferring markup language information and, particularly, hypertext markup language (HTML) documents, to the client digital data processors **12, 14**. In alternative embodiments of the invention, information server **20** can comprise any other such server capable of supplying an applet to the client digital data processors **12, 14** in response to requests by them.

Id., col. 4 ll. 57–65. Fisher-Rosemount’s corresponding proposed construction reads: “an application, such as a hypertext transfer protocol (HTTP) server, that is capable of supplying applets to a requesting client.”

Lexicography only arises when the patent drafter “clearly, deliberately, and precisely define[s] the term.” *Sinorgchem Co., Shandong v. Int’l Trade Comm’n*, 511 F.3d 1132, 1136 (Fed. Cir. 2007). Clear definitions are usually set off by quotation marks or are marked by the word “is.” *Id.* Here, instead of clearly and specifically defining these terms, the patent is simply listing some of the capabilities of the disputed terms. Describing what an information server or information client can do is different from describing what they must do as a matter of definition.

Furthermore, Fisher-Rosemount relies on these select passages to the exclusion of the rest of the specification, which consistently describes information client and information server more broadly in the context of establishing communications over a network. *See, e.g.*, ’195 Patent, col. 4 ll. 47–51 (“In addition, [digital data processors] execute information client/server software that configures them to carry on high-level communications, particularly, over the Internet.”); *id.*, col. 4 ll. 52–56 (“More particularly, in the illustrated embodiment, server digital data processor **16**

includes information server **20** responsible for establishing communications over network **18** with information clients executing on the client digital data processors **12, 14.**”); *id.* cols. 4–5 ll. 66–2 (“The information server **20** establishes communications with the client digital data processors **12, 14** and, particularly, their respective information clients in the conventional manner known in the art.”).

Fisher-Rosemount cites *Network Commerce, Inc. v. Microsoft Corp.*, 422 F.3d 1353 (Fed. Cir. 2005), which, in construing “download component,” refused to simply combine definitions of the individual terms “download” and “component.” *Id.* at 1360. The Special Master, however, does not recommend combining the dictionary definitions of “information” and “server” or “client.” Instead, the Special Master reasonably determined a person of ordinary skill in the art could read these terms in the context of the specification and understand the scope of the invention.

In the end, Fisher-Rosemount seeks to import a list of capabilities provided in the specification into the claims as strict limitations. The Court sees no reason to confine the claims in this way. The jury should have little problem consulting the specification, understanding the passages describing the capabilities of information clients and information servers, and applying those listed capabilities when reading information client and information server onto the accused device.

The Court agrees with the Special Master’s recommendation, and Fisher-Rosemount’s objection is OVERRULED.

5. “virtual machine environment”

The Special Master recommended that construction is unnecessary for this term. As with information client and information server, Fisher-Rosemount argues “virtual machine environment”

is defined in the specification and that Invensys should be bound by that definition. Specifically, Fisher-Rosemount cites the specification's statement that a virtual machine environment "compris[es] the Java programming language run-time platform and Java interpreter" and is "platform independent (i.e., hardware-independent, operating system-independent and windows system independent)." '195 Patent, col. 5 ll. 32–45. Fisher-Rosemount further points out the parties agree an "applet" is "intermediate or executable code that is suitable for interpretation or execution within the virtual machine environment that is hardware-independent, operating system-independent and windows system independent." *Id.*, col. 5 ll. 25–29. Fisher-Rosemount therefore concludes "virtual machine environment" must be construed as limited to running applets and proposes the following definition: an "application that interprets and executes hardware-independent, operating system-independent and windows system independent intermediate or executable code."

As a an initial matter, virtual machine environments have been known in the art of computer science since the 1970s. *See* Second Stephenson Decl. ¶ 43. Skilled artisans at the time of the invention in the mid-1990s would have understood the scope of virtual machine environments. Instead of relying on this understood meaning, however, Fisher-Rosemount seeks construction of the term. The Special Master found no construction is necessary, and the Court agrees for multiple reasons. First, although Fisher-Rosemount argues Invensys acted as its own lexicographer for the term virtual machine environment, Fisher-Rosemount cites no language in the specification evincing a clear and deliberate intent to define the term virtual machine environment. Instead, Fisher-Rosemount pieces together different sections of the specification and draws its own conclusion as to what the definition should be.

Even if the Court thought construction were necessary, Fisher-Rosemount's reasoning is faulty. Specifically, Fisher-Rosemount relies on the agreed definition of applet as running in a virtual machine environment to mean all virtual machine environments must, as a matter of definition, execute applets. The latter does not follow from the former. Moreover, the agreed definition of applet indicates it is intermediate or executable code suitable for "interpretation *or* execution within the virtual machine environment." Fisher-Rosemount's proposed construction, however, requires both interpretation *and* execution of the code. Fisher-Rosemount fails to explain this added limitation.

Inserting Fisher-Rosemount's proposed construction for virtual machine environment into the agreed definition of applet demonstrates its flaws and also results in an unwieldy definition for a jury: "intermediate or executable code suitable for interpretation or execution with the application that interprets and executes hardware-independent, operating system-independent and windows system independent intermediate or executable code that is hardware-independent, operating system-independent and windows system independent." Fisher-Rosemount fails to account for the inconsistent and confusing results of its construction.

In sum, the Court rejects Fisher-Rosemount's proposed construction as an attempt to limit virtual machine environments to a specific embodiment described in the patent. "Virtual machine environment" has a well understood meaning in the art, and limitation of terms with a well understood meaning in the art to specific embodiments is appropriate only if the inventors exhibit a clear intent to redefine the term. *See Thorner v. Sony Computer Entm't Am. LLC*, 669 F.3d 1362, 1365 (Fed. Cir. 2012). Fisher-Rosemount fails to cite language in the specification demonstrating such intent.

The Court agrees with the Special Master's recommendation that no construction is necessary, and Fisher-Rosemount's objection is OVERRULED.

6. “digital data processor is disposed remotely from the portable computer”

The Special Master recommended that construction is unnecessary for this term. Fisher-Rosemount objects to this recommendation and argues “disposed remotely” must be construed as located outside the factory or facility. For this limitation, Fisher-Rosemount relies on statements in the “Background of the Invention” section of the specification, which is the only portion of the patent discussing the term “remotely.” Specifically, the Background provides:

Although modern process control systems, such as the I/A Series systems, have proven quite successful, to date they have provided only limited remote access capabilities. Thus, while numerous operator workstations may reside within the factory or facility in which the control/sensing devices are disposed, it has traditionally proven difficult to access and control those devices outside those areas.

Remote access and control of processes is desirable for a number of purposes. A plant manager who is “on the road,” for example, may wish to monitor the plant processes while travelling. By way of further example, the manufacturer of process control equipment may require remote access to a plant's control/sensing devices in order to provide technical support.

'656 Patent, col. 1 ll. 54–67.

Based on this language, Fisher-Rosemount contends Invensys defined “remotely” as located outside the facility, but Fisher-Rosemount overstates the implications of these two paragraphs. Instead of demonstrating a deliberate intent to define “remotely,” the above-quoted language merely expresses some of the potential practical benefits of the invention as part of the background. *See Brookhill-Wilk 1, LLC v. Intuitive Surgical, Inc.*, 334 F.3d 1294, 1301 (Fed. Cir. 2003) (“Advantages described in the body of the specification, if not included in the claims, are not per se limitations to the claimed invention.”) (internal quotation omitted). The language does not clearly disavow remote

access and control of processes while still inside the facility. *Home Diagnostics, Inc. v. LifeScan, Inc.*, 381 F.3d 1352, 1358 (Fed. Cir. 2004) (“Absent a clear disavowal in the specification or the prosecution history, the patentee is entitled to the full scope of its claim language.”). The patentee described examples of remote access occurring while outside the facility, but while remote access may often equate to “outside the facility,” that does not rule out remote access while still within the facility.

While the Court appreciates the word “remotely” is not the most precise way to define the relationship of the digital data processor and the portable computer, the Court also does not think it is so difficult as to cause significant problems for a jury. Moreover, there is no indication a person of ordinary skill in the art would have not have been able to understand the word “remotely” as used in the ’656 Patent. The Special Master was correct to conclude no construction is necessary, and Fisher-Rosemount’s objection is OVERRULED.

7. “JAVA/API application for transferring information between the JAVA application and the API”

The Special Master recommended that construction is unnecessary for this claim phrase. Fisher-Rosemount objects and argues the term “JAVA/API application” is ambiguous to the point of being indefinite. Even more precisely, Fisher-Rosemount does not take issue with JAVA, JAVA application, or API standing alone, but rather considers the “/” mark to be the source of the indefiniteness. *See* Pl.’s Opening Claim Construction Br. [#65] at 29.

The term “JAVA/API application” only appears in claim 15 of the ’944 Patent. The claim reads in its entirety:

15. A system according to claim 7, wherein the command processor comprises

- A. a JAVA application executing on the server digital data processor for transferring information at least one of to and from the network,
- B. a non-JAVA application program interface (API) for transferring information at least one of to and from the process control apparatus, and
- C. a JAVA/API application for transferring information between the JAVA application and the API.

'944 Patent, col. 12 ll. 23–32.

The Court agrees a “/” mark has multiple potential meanings, and the context is key to determining the applicable meaning. *See Phillips*, 415 F.3d at 1314 (“To begin with, the context in which a term is used in the asserted claim can be highly instructive.”). The term “JAVA/API application,” when read in the context of claim 15, is clear: it is a software application that transfers information between the claimed JAVA application recited in limitation A and the non-JAVA API recited in limitation B. The term informs, with reasonable certainty, those skilled in the art about the scope of the invention, and as the Special master recommended, no construction is necessary.

Fisher-Rosemount’s objection is **OVERRULED**.

Conclusion

The parties’ objections to the Special Master’s recommended constructions are **OVERRULED**, and the Special Master’s recommended constructions are **ACCEPTED**.


Accordingly,

IT IS ORDERED that Fisher-Rosemount Systems, Inc. and Emerson Electric Co.’s Objections [#87] are **OVERRULED**;

IT IS FURTHER ORDERED that Invensys Systems, Inc.’s Objections [#88] are **OVERRULED**;

IT IS FINALLY ORDERED that the Report and Recommendation of the Special Master [#84] is ACCEPTED.

SIGNED this the 18th day of March 2015.



SAM SPARKS
UNITED STATES DISTRICT JUDGE